



HAL
open science

Analysis of the Robustness of Critical Infrastructures Within a Multistate Systems-of-Systems Framework in the Presence of Epistemic Uncertainties

E Ferrario, N Pedroni, Enrico Zio

► **To cite this version:**

E Ferrario, N Pedroni, Enrico Zio. Analysis of the Robustness of Critical Infrastructures Within a Multistate Systems-of-Systems Framework in the Presence of Epistemic Uncertainties. ICVRAM & ISUMA 2014, Jul 2014, Liverpool, United Kingdom. pp.715 - 724, 10.1061/9780784413609.073 . hal-01108198

HAL Id: hal-01108198

<https://centralesupelec.hal.science/hal-01108198>

Submitted on 22 Jan 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Analysis of the robustness of critical infrastructures within a multistate systems-of-systems framework in the presence of epistemic uncertainties

E. Ferrario¹, N. Pedroni¹ and E. Zio^{1,2}

¹Chair on Systems Science and the Energetic Challenge, European Foundation for New Energy - Electricité de France, at École Centrale Paris - Supelec, France

²Department of Energy, Politecnico di Milano, Italy

ABSTRACT

In this paper, we look at the robustness of connected critical infrastructures under a systems-of-systems framework taking into account i) the dependencies and interdependencies among the components of a critical infrastructure and between different critical infrastructures, respectively; ii) the variability of the performance of each component by means of a multistate model; iii) the epistemic uncertainty in the transition probability between different components states by means of probability intervals. We adopt the Goal Tree Success Tree – Dynamic Master Logic Diagram for system modelling and we perform a quantitative assessment of the systems-of-systems performance by Monte Carlo simulation. We illustrate the approach by way of a simplified case study consisting of two interdependent infrastructures (electric power system and a gas network) and a supervisory control and data acquisition system connected to the gas network.

1. INTRODUCTION

Critical infrastructures, e.g., transportation, electric power, water, oil, gas and communication systems, interact on the basis of complex relationships that cross the single infrastructure boundary, increasing, in this way, the risk of their failure: actually, a failure of an infrastructure can significantly impact another one. For example, the widespread power electric blackout occurred in the Midwest and Northeast United States and Ontario, Canada, on August 2003, affected the serviceability of the water system at Cleveland, OH, due to the lack of power needed to the water pumping stations to operate [Adachi and Ellingwood, 2008]. In addition, critical infrastructures are getting more and more dependent on information technologies that, on one hand, provide control and support to them increasing their efficiency, but, on the other hand, create new vulnerabilities [Nozick et al., 2005]. Understanding these interdependences between infrastructure systems is fundamental to the well-functioning of these “systems of systems”.

Under a systems-of-systems framework of analysis, we wish to estimate the systems-of-systems performance, in terms of robustness, considering the dependencies and interdependencies among the components of a critical infrastructure and between different critical infrastructures, respectively. For a more realistic representation, we adopt a multistate model where different degrees of damage of the individual

components are contemplated [Ferrario and Zio, 2013]: transitions between these different states of damage occur stochastically. In addition, we take into account the epistemic uncertainty affecting the transition probabilities between different states, due to the lack of knowledge and information on the system: actually, in many reliability assessments of safety-critical infrastructures, few observations of the system failure behavior are available and thus it is difficult to estimate their levels of damage and the precise values of the corresponding transition probabilities [Sallak et al., 2013].

For illustration purposes, we adopt the case study proposed by [Nozick et al., 2005], consisting of two interdependent infrastructures (gas and electric power networks) and a supervisory control and data acquisition (SCADA) system connected to the gas network. As a measure of the robustness of the system, we determine the steady-state probability distributions of the supply of gas and electricity at the demand nodes.

We propose a hierarchical model description by Goal Tree Success Tree – Dynamic Master Logic Diagram (GTST-DMLD) [Hu and Modarres, 1999], extending its representation characteristics to evaluate the physical flows of gas and electricity through the interdependent infrastructures. We adopt intervals to describe the epistemic uncertainty in the probabilities of transition between different components states and we use interval analysis to calculate the (uncertain) probabilities of the states of all the components of the critical infrastructures [Buckley, 2004]. Finally, we employ Monte Carlo simulation [Kalos, 1986; Zio, 2013] for the probabilistic evaluation of systems-of-systems performance.

The remainder of the paper is organized as follows. In Section 2, the case study is presented; in Section 3, the systems-of-systems modelling by Goal Tree Success Tree - Dynamic Master Logic Diagram is illustrated; in Section 4, details of the procedural steps to evaluate the systems-of-systems performance under epistemic uncertainty are given; in Section 5, the results of the analysis are shown and commented; in Section 6, conclusions are provided.

2. CASE STUDY

The case study is taken from [Nozick et al., 2005] and deals with two interconnected infrastructures, i.e., a natural gas distribution network and an electricity generation/distribution network (Figure 1, solid and dashed lines, respectively). The gas distribution network is supported by a supervisory control and data acquisition (SCADA) system (Figure 1, dotted lines). The objective of these interconnected systems of systems is to provide the necessary amount of gas and electricity (hereafter also called “product”) to four demand nodes (end-nodes), namely D1 and D2 (gas) and L1 and L2 (electricity).

The gas distribution network, supplied by two sources of gas (namely, S1 and S2, that are connected to the network by arcs S1_DS1 and S2_DS2, respectively) provides gas to the end-nodes D1 and D2 and to two nodes of the electricity network (E1 and E2). Once the gas enters into nodes E1 and E2, it is transformed into electrical energy that flows through arcs E1_G1 and E2_G2 (representing the electric power generation stations) to supply the end-nodes of electricity (L1 and L2); notice

that the demand L2 can be supplied by both electrical generations E1_G1 and E2_G2. The assumption is made that the gas-electricity transformation occurs with a constant coefficient, i.e., 100 cu. ft. of natural gas produces 1 MWh of electricity [Nozick et al., 2005].

A SCADA system controls the gas flow through arcs a_b, b_c, c_d and d_e. It is assumed that: i) the SCADA has two core subsystems controlling different sets of arcs (in particular, the first one – SUB1 – refers to links a_b and b_c, whereas the second one – SUB2 – controls arcs c_d and d_e); ii) the SCADA is always provided by electric power [Nozick et al., 2005].

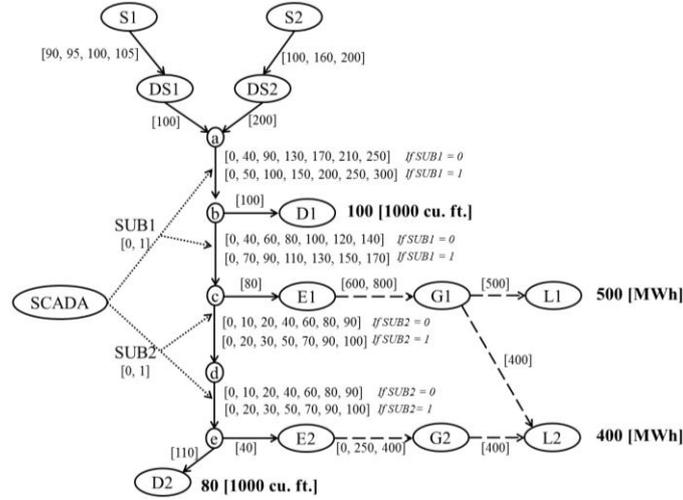


Figure 1: Interdependent gas (solid-black lines) and electric (dashed lines) infrastructures and SCADA system (dotted lines) [Nozick et al., 2005].

The capacities of the arcs of the gas and electricity network (determining the maximum flows of gas or electricity supported by the arc) can be deterministic (i.e., fixed constant values) or stochastic (i.e., randomly evolving in time) (Figure 1, values in the square brackets). The stochastic capacities give rise to a multistate model that reflects the possibly different degrees of damage of the arc. On the contrary, the SCADA system state is defined by a binary random variable, whose values 1 and 0 represent its complete and partial functioning, respectively. For example, when the state of the SCADA subsystem SUB1 (controlling arcs a_b and b_c) is 0, the capacity of these arcs decreases because of the incorrect information provided by the SCADA subsystem (even if the arcs are not subject to a direct damage). On the basis of the two states of the SCADA subsystems, two different vectors of capacities are identified for each arc a_b, b_c, c_d and d_e: as illustrated in Figure 1, the first vector is used when the corresponding SCADA subsystem is in state 0, whereas the second one is utilized when the SCADA subsystem is in state 1. In the following, we generically denote the value of the state of a component (i.e., the capacity of the arcs) as $\zeta_{comp,i}$, $i \in \{1,2,\dots,S_{comp}\}$, where the subscript ‘comp’ indicates the component of interest and S_{comp} is the total number of states for that component. For example, component S1_DS1 has $S_{S1_DS1} = 4$ possible states: $\zeta_{S1_DS1,1} = 90$ [1000 cu. ft.], $\zeta_{S1_DS1,2} = 95$ [1000 cu. ft.], $\zeta_{S1_DS1,3} = 100$ [1000 cu. ft.], $\zeta_{S1_DS1,4} = 105$ [1000 cu. ft.]. The total number of components in the systems of systems is referred to as N_{comp} .

Changes in the arc capacities are due to random failures or recovery actions. The state transitions over time are modeled by Markov and semi-Markov processes as in [Nozick et al., 2005]. Semi-Markov processes are adopted to represent the evolution of the capacities of the gas supply links (S1_DS1 and S2_DS2), whereas Markov processes are used for all the others arcs. Both Markov and semi-Markov processes for a generic component ‘*comp*’ are defined by a transition probability matrix $\underline{\underline{P}}_{comp} = \{p_{ij} : i, j = 1, 2, \dots, S_{comp}\}$, where p_{ij} is the one-step probability of transition from state i to state j . In addition, the semi-Markov processes are characterized by continuous probability distributions for the holding time T_{comp}^{ij} , i.e., for the time of residence in state i before performing a transition to state j .

Differently from [Nozick et al., 2005], we take into account the epistemic uncertainty affecting the transition probabilities and the holding time distributions of the Markov and semi-Markov processes, respectively. In particular, intervals, $[\underline{p}_{ij}, \bar{p}_{ij}]$, $i, j = 1, \dots, S_{comp}$, (instead of fixed constant values) are used to describe the state transition probabilities for both Markov and semi-Markov processes (matrices $\underline{\underline{P}}_{comp}$, $comp = S1_DS1, S2_DS2, a_b, b_c, c_d, d_e, SCADA, E1_G1$ and $E2_G2$, in Figure 2 with respect to the states defined in Figure 1). The holding time distributions for the semi-Markov process are considered normal with epistemically-uncertain mean (described by an interval) and fixed standard deviation (matrices $\underline{\underline{T}}_{comp}$, $comp = S1_DS1, S2_DS2$, in Figure 2).

<p>comp = S1_DS1 (Semi-Markov)</p> $\underline{\underline{P}}_{comp} = \begin{array}{c ccc} & 0 & 1 & 0 \\ & 0 & 0 & 1 \\ \hline [0.002; 0.008] & [0.002; 0.008] & 0 & [0.998; 1] \\ [0.002; 0.008] & [0.002; 0.008] & [0.998; 1] & 0 \end{array}$ $\underline{\underline{T}}_{comp} = \begin{array}{c ccc} & - & N([2; 6], 1) & - \\ & - & - & N([2; 6], 1) \\ \hline N([7; 13], 3) & N([7; 13], 3) & - & N([17; 23], 2) \\ N([7; 13], 3) & N([7; 13], 3) & N([17; 23], 2) & - \end{array}$	<p>comp = S2_DS2 (Semi-Markov)</p> $\underline{\underline{P}}_{comp} = \begin{array}{c ccc} & 0 & 1 & 0 \\ & [0; 0.02] & 0 & [0.98; 1] \\ \hline [0; 0.02] & [0.98; 1] & 0 & \end{array}$ $\underline{\underline{T}}_{comp} = \begin{array}{c ccc} & - & N([2; 6], 1) & - \\ & N([7; 13], 3) & - & N([2; 6], 1) \\ \hline N([7; 13], 3) & N([17; 23], 2) & - & \end{array}$
<p>comp = a_b, b_c, c_d, d_e (Markov)</p> $\underline{\underline{P}}_{comp} = \begin{array}{c ccccccc} [0.04; 0.06] & [0.04; 0.06] & [0.04; 0.06] & [0.04; 0.06] & [0.1; 0.3] & [0.1; 0.3] & [0.3; 0.5] \\ [0.04; 0.06] & [0.04; 0.06] & [0.04; 0.06] & [0.1; 0.3] & [0.04; 0.06] & [0.1; 0.3] & [0.3; 0.5] \\ [0.04; 0.06] & [0.04; 0.06] & [0.04; 0.06] & [0.1; 0.3] & [0.1; 0.3] & [0.04; 0.06] & [0.3; 0.5] \\ [0.04; 0.06] & [0.1; 0.3] & [0.04; 0.06] & [0.1; 0.3] & [0.04; 0.06] & [0.04; 0.06] & [0.3; 0.5] \\ [0.04; 0.06] & [0.1; 0.3] & [0.04; 0.06] & [0.1; 0.3] & [0.1; 0.3] & [0.04; 0.06] & [0.3; 0.5] \\ [0.04; 0.06] & [0.1; 0.3] & [0.04; 0.06] & [0.1; 0.3] & [0.04; 0.06] & [0.04; 0.06] & [0.3; 0.5] \\ [0.0005; 0.0015] & [0.0005; 0.0015] & [0.0005; 0.0015] & [0.0015; 0.0025] & [0.0015; 0.0025] & [0.002; 0.004] & [0.985; 0.995] \end{array}$	
<p>comp = SCADA (Markov) (states of SUB1 and SUB2: 0 0, 0 1, 1 0, 1 1)</p> $\underline{\underline{P}}_{comp} = \begin{array}{c cccc} [0.7; 0.9] & [0.03; 0.05] & [0.03; 0.05] & [0.02; 0.22] \\ [0.05; 0.15] & [0.3; 0.5] & [0.2; 0.4] & [0.1; 0.3] \\ [0.05; 0.15] & [0.2; 0.4] & [0.3; 0.5] & [0.1; 0.3] \\ [0.0005; 0.0007] & [0.0001; 0.0003] & [0.0001; 0.0003] & [0.998; 1] \end{array}$	<p>comp = E2_G2 (Markov)</p> $\underline{\underline{P}}_{comp} = \begin{array}{c ccc} [0.1; 0.3] & [0.05; 0.15] & [0.6; 0.8] \\ 0 & [0.1; 0.3] & [0.7; 0.9] \\ [0.0004; 0.0006] & [0.0004; 0.0006] & [0.998; 1] \end{array}$ <p>comp = E1_G1 (Markov)</p> $\underline{\underline{P}}_{comp} = \begin{array}{c cc} [0.05; 0.15] & [0.89; 0.91] \\ [0; 0.002] & [0.998; 1] \end{array}$

Figure 2: Holding time distributions (matrices $\underline{\underline{T}}_{comp}$) for the arcs described by semi-Markov processes and state transition probability matrices ($\underline{\underline{P}}_{comp}$) for the arcs described by Markov and semi-Markov processes.

In the present work, the demand nodes are not given the same importance: in

particular, D1 is more important than L1; on its turn, L1 is more important than both D2 and L2 (which instead are equally important). These assumptions are made to illustrate and motivate the repartition of electricity and gas flows in the network, as represented in the Goal Tree Success Tree - Dynamic Master Logic Diagram given in the next Section 3.

The objective of the analysis is to determine the cumulative distribution functions of the product delivered to the demand nodes (i.e., D1, D2, L1, L2) at steady state. Since the state transition probabilities of the network components are affected by epistemic uncertainty, described by intervals, $[\underline{p}_{ij}, \bar{p}_{ij}]$, $i, j = 1, \dots, S_{comp}$, the corresponding component steady-state probabilities are also affected by epistemic uncertainty and represented by intervals of possible values, $[\Pi_{\min}^{comp,i}, \Pi_{\max}^{comp,i}]$, $i = 1, \dots, S_{comp}$. As a consequence, a set of cumulative distribution functions corresponding to the set of possible steady-state probabilities within the intervals $[\Pi_{\min}^{comp,i}, \Pi_{\max}^{comp,i}]$, $i = 1, \dots, S_{comp}$, is obtained for each demand node.

3. SYSTEMS-OF-SYSTEMS MODELLING

3.1. GTST-DMLD: basic concepts

The Goal Tree Success Tree – Dynamic Master Logic Diagram (GTST-DMLD) is a goal-oriented method based on a hierarchical framework [Hu and Modarres, 1999]. It gives a comprehensive knowledge of the system describing the complex physical systems in terms of functions (qualities), objects (parts) and their relationships (interactions). The first part is developed by the Goal Tree (GT), the second one by the Success Tree (ST) and the third one by the DMLD [Hu and Modarres, 1999].

In extreme synthesis, the GT identifies the hierarchy of the qualities of the system decomposing the objective of the analysis, i.e., the goal, into functions that are in turn divided into other functions and so on by answering the question “how” they can attain the parent function (looking from top to bottom of the hierarchy) and “why” the functions are needed (looking from bottom to top of the hierarchy) [Brissaud et al., 2011]. The ST represents the hierarchy of the objects of the system from the whole system to the parts necessary to attain the last levels of the GT. This hierarchy is built identifying the elements that are “part of” the parent objects [Brissaud et al., 2011]. The DMLD is an extension of the Master Logic Diagram (MLD) [Hu and Modarres, 1999] to model the dynamic behavior of a physical system. It identifies the interactions between parts, functions and parts and functions, in the form of a dependency matrix and it adds the dynamic aspect by introducing time-dependent fuzzy logic [Hu and Modarres, 1999].

3.2. GTST-DMLD for interconnected networked infrastructures

In this Section, we adapt the GTST-DMLD presented in Section 3.1 for the representation of interconnected networked infrastructures. In particular, we introduce new concepts in order to highlight in the diagram not only the dependency relations between the components, but also the ways in which the flows of gas and electricity are partitioned into the network on the basis of i) the importance of the demand nodes, ii) the amount of product necessary to satisfy each demand, iii) the constraints of the arc capacities, and iv) the information provided by the SCADA

system. In the following, first we explain the notation adopted in the GTST-DMLD and, then, we apply it to the case study of interest.

First of all, since in the present work we are interested in analyzing the flows passing through the network, the input and output of an arc are flows and the output is (generally) the sum of the flow inputs. This situation is represented by a “+” in the middle of an “AND” gate, as shown in the example of Figure 3 a. where the flows of arcs A and B enter into arc C.

With respect to the dependency relations, we distinguish between three main types: *direct*, *indirect* and *constraint-based* dependencies, as illustrated in Figures 3 and 4. The first ones, pictorially represented by dots and hereafter called "*dot-dependencies*", express the fact that the product of the element on the bottom passes straightly into the element on the top. The indirect dependencies, represented by hexagons and called hereafter "*hexagon dependencies*", are instead needed for the optimal allocation of the product in the network: for example, they are used to describe those cases where the flow exceedance in an arc can be better partitioned into another arc that is not directly connected to it but that shares one of the inputs (see the example of Figure 3 b). Finally, the constraint-based dependencies, depicted by triangles and hereafter called "*triangle-dependencies*", are employed to take into account some physical constraints posed by the problem, like the maximum flow required by a demand node.

For clarity of illustration, in Figure 3, examples of two types of dot- and hexagon-dependencies are given, with respect to different graph representations. Figure 3 a. shows the dependence of arc C on two input arcs A and B: arc C receives all the input products from A and B; this complete direct dependence is depicted by a *black dot*. Figures 3 b. and c. describe the same "physical" situation (i.e., an input arc A and two output arcs B and C), but with different relative importances of the arcs. Two different cases are illustrated. In the first case (Figure 3 b.), arc B is more important than C: thus, in this situation, the flow from A supplies first arc B until its demand is satisfied, and then arc C. In the second case (Figure 3 c.), arcs B and C are equally important: thus, the input flow (A) is divided into equal parts on the basis of the number of output arcs (i.e., two in this example). In the case of Figure 3 b., the flow that enters in C is given by the difference between the entire flow from A and the flow given to B; to represent and compute this difference in the DMLD, the hexagon-dependency is adopted to correct the black dot-dependency from arc A to arc C (in fact, it is impossible that the entire flow of A enters at the same time in the arcs B and C as expressed by the black dot-dependency). The *white hexagon* assumes the value of the flow in B with a negative sign; this value is then summed to the initial flow of A to obtain the flux to C. The flow given to B can be the entire flow of A or a lower value depending on the constraints and arc capacity (see the following example in Figure 4). In the case of Figure 3 c., the flow from A is divided into equal parts: this condition is represented by a *grey dot*. However, this equal partition of the flow may not represent the optimal one, since some output arcs may require less flow than the one allocated according to this criterion. Thus, to optimize the repartition of the flow, hexagon-dependencies are adopted: they are directed from an output arc to all the other output arcs that share the same input. In this case, the “surplus flow” is a

positive quantity and it is represented by a *grey hexagon* (to distinguish it from the “negative” white hexagon of the example in Figure 3 b).

Notice that the graph representation of Figures 3 b. and 3 c. are identical; however, the partition of the flux from A is completely different in the two cases: this means that the graph representation alone cannot be used to describe the repartition of the flows in the network according to different criteria. On the contrary, the DMLD can capture and depict this aspect, that is useful in the quantitative evaluation of the system performance.

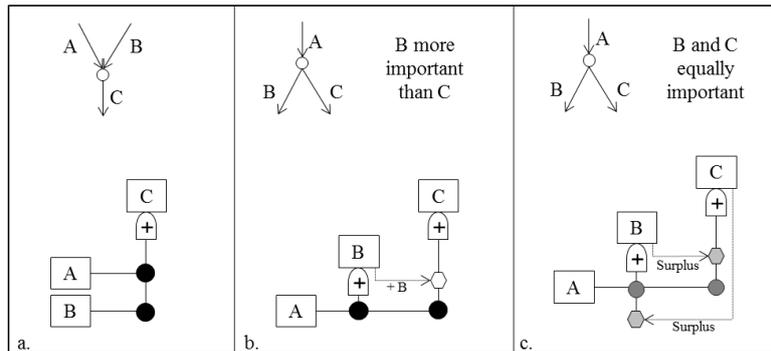


Figure 3: Examples of dot- and hexagon-dependencies with respect to possible graph representations.

In Figure 4, examples of two types of triangle-dependencies are given, with respect to different possible graph representations. Figure 4 a. depicts the same situation as Figure 3 a. with an additional arc D whose behavior impacts on the state of arc C (however, notice that D is not an input to C). This dependency is represented by a grey triangle and it means that the output of C can be modified on the basis of the state of arc D. In the present case study, this constraint-based dependency is used to model the SCADA system that can decrease the actual flow of the controlled arc if it is in a damage state. Figure 4 b. represents the same situation of Figure 3 c. with the addition of another arc (D) sequential to arc C. In this case, the capacity (or the demand) of arc D can limit the amount of flow in input to arc C. This constraint is represented in the DMLD by a black triangle and it is needed to control the input flow partitioned in different arcs and guarantee that it is not higher than necessary.

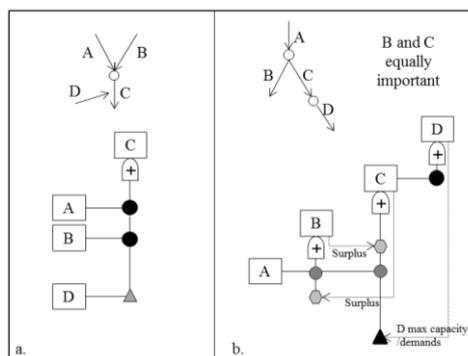


Figure 4: Examples of triangle-dependencies with respect to possible graph representations.

Finally, another type of constraint is taken into account, i.e., the one related to the capacity of the arcs: when the flow in input to an arc is higher than the capacity of

the arc itself, the output flow will be equal to the capacity of the arc. The arc capacity can be deterministic or stochastic and in the GTST-DMLD it is represented by a grey or dot-filled rectangular, respectively (see Figure 5).

In Figure 5, the GTST-DMLD of the case study of Section 2 is shown.

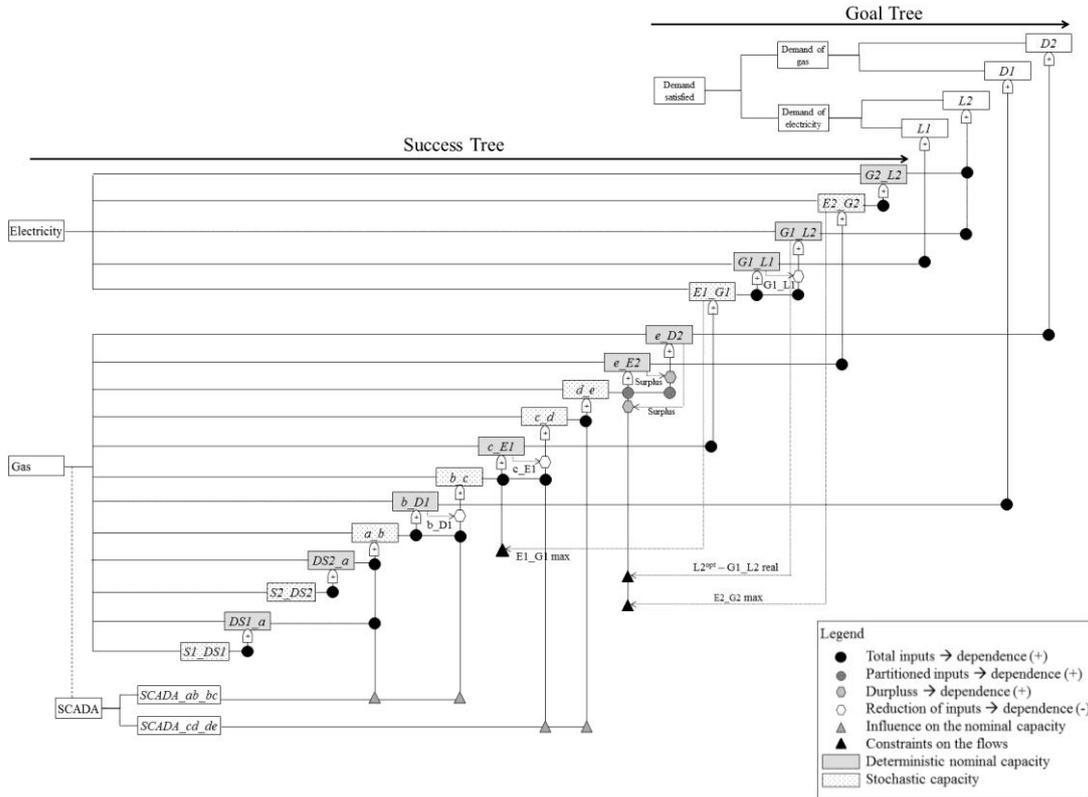


Figure 5: GTST-DMLD of the case study with respect to the graph of Figure 1.

The Goal Tree, on the top, represents the main goal of the systems of systems that is related to the supply of the demands of gas and electricity. The objective is achieved if the corresponding nodes D1, D2, L1 and L2 receive the required amount of gas and electricity, respectively. In the present case study, we limit the analysis to the last level of the GT, i.e., we analyze the performance of each demand, without investigating a global indicator of the systems of systems.

The Success Tree is composed by the *main* hierarchies of gas and electricity networks (that directly provide the demand nodes with gas and electricity to achieve the goal function) and by the *support* hierarchy of the SCADA system (that is needed for the control of the gas network and, therefore, it is not directly involved in the achievement of the goal function). Given its support role, it is represented in a parallel dashed branch connected to the gas hierarchy.

The DMLD is represented by the relationships between objects of the ST or between objects of the ST and functions of the GT. It allows determining the goal function by the evaluation of all the dependencies from the bottom to the top of the diagram, following the rules explained above for the dot-, hexagon- and triangle-dependencies. For example, arc a_b depends on two arcs, DS1_a and DS2_b, connected by black dot-dependencies (Figure 5). Thus, the output of a_b is given by

the sum of the corresponding input values, i.e., $DS1_a + DS2_b$. This value may, then, be modified by the triangle constraint of the SCADA system and by the (stochastic) capacity of arc a_b itself.

4. SYSTEMS-OF-SYSTEMS PERFORMANCE EVALUATION

In this Section, we illustrate the algorithm adopted to evaluate the performance of the systems of systems described in Section 2, in the presence of epistemic uncertainties in the components state transition probabilities (represented by intervals). As already mentioned in Section 2, the system performance is quantified in terms of the steady-state probability distributions of the product delivered at the demand nodes. The algorithm consists of the following three main steps:

1. Processing the epistemic uncertainties by interval analysis: this step leads to the evaluation of the intervals of the steady-state probabilities, $[\Pi_{\min}^{comp,i}, \Pi_{\max}^{comp,i}]$, $i = 1, 2, \dots, S_{comp}$, for the states of each component ($comp = 1, 2, \dots, N_{comp}$) of the systems of systems.
2. Evaluation of the systems-of-systems performance by Monte Carlo simulation: this step leads to the determination of a set of cumulative distribution functions (CDFs) of the product delivered at each demand node at steady state, one for each possible combination of steady-state probabilities ranging within the intervals $[\Pi_{\min}^{comp,i}, \Pi_{\max}^{comp,i}]$, $i = 1, 2, \dots, S_{comp}$, (found at step 1. above).
3. Post-processing the results obtained at the previous step 2: this step leads to the identification of two extreme upper and lower CDFs that bound the set of CDFs produced at step 2. above.

5. RESULTS

Table 1 reports the upper and lower probabilities that the product delivered at steady state to the demand nodes, D1, D2, L1 and L2, exceeds the following threshold values: $d_1^* = 95$ [1000 cu. ft.], $d_2^* = 75$ [1000 cu. ft.], $l_1^* = 475$ [MWh] and $l_2^* = 375$ [MWh] (i.e., the probabilities that the corresponding demands are satisfied).

Table 1: lower and upper probabilities that the product delivered to the demand nodes (D1, D2, L1 and L2) exceeds the corresponding requested threshold value.

D1 $\geq d_1^* = 95$ [1000 cu. ft.] [lower, upper]	D2 $\geq d_2^* = 75$ [1000 cu. ft.] [lower, upper]	L1 $\geq l_1^* = 475$ [MWh] [lower, upper]	L2 $\geq l_2^* = 375$ [MWh] [lower, upper]
[0.971, 1]	[0.450, 0.780]	[0.963, 1]	[0.929, 0.992]

It can be seen that, in general, the probability of satisfying demand nodes D1 and L1 is higher than for nodes D2 and L2: their threshold values are satisfied, in the worst case, with probability equal to 0.971 and 0.963, respectively. On the contrary, node D2 is the least supplied: the upper and lower probabilities that the product delivered to it exceeds the corresponding threshold value are low, i.e., 0.450 and 0.780, respectively. This is due to the fact that node D2 can be satisfied by only one path that presents high epistemic uncertainty in the arc capacities (a_b , b_c , c_d and d_e). On the contrary node L2, is satisfied with probability between 0.929 and 0.992 even if it is the farthest node from the input sources (and, thus, more affected by uncertainties in the arc capacities): this is due to the presence of two redundant paths that allow its supply by arcs E1_G1 and E2_G2.

6. CONCLUSIONS

In this paper, we have considered systems of systems made of interdependent infrastructures and proposed a modelling framework to evaluate the robustness taking into account i) the dependencies and interdependencies among the components of a critical infrastructure and between different critical infrastructures, respectively, ii) the variability in the states of the components (by adopting a multistate model), and iii) the epistemic uncertainty in the transition probabilities between different components states (by interval analysis).

For exemplification, we have performed an analysis of interconnected gas and electricity networks, with a supervisory control and data acquisition (SCADA) system connected to the gas network, by using the Goal Tree Success Tree – Dynamic Master Logic Diagram for system modeling and Monte Carlo simulation for the quantitative evaluation of performance at steady state. The results obtained can help to improve the global systems-of-systems performance by improving the structural response of specific arcs that more easily turns into damage states or by developing a more redundant network that allows the supply of the product from different paths.

REFERENCES

- Adachi, T., and Ellingwood, B. (2008). Serviceability of earthquake-damaged water systems: Effects of electrical power availability and power backup systems on system vulnerability. *Reliab Eng Syst Saf*, 93(1), 78-88.
- Barry L.N. (1995). *Stochastic modeling: analysis and simulation*. McGraw-Hill, New York.
- Brissaud, F., Barros, A., Bérenguer, C., Charpentier, D. (2011). Reliability analysis for new technology-based transmitters. *Reliab Eng Syst Saf*, 96, 299-313.
- Buckley, JJ. (2004). *Fuzzy probabilities and fuzzy sets for web planning*. Series: Studies in Fuzziness and Soft Computing, Vol. 135. Springer, Berlin.
- Ferrario, E., and Zio, E. (2013). Goal Tree Success Tree - Dynamic Master Logic Diagram and Monte Carlo Simulation for the Safety and Resilience Assessment of a Multistate System of Systems. Accepted for publication in *Eng Struct*.
- Hu, YS., and Modarres, M. (1999) Evaluating system behavior through Dynamic Master Logic Diagram (DMLD) modeling. *Reliab Eng Syst Saf*, 64, 241-269.
- Kalos, M.H., and Whitlock, PA. (1986). *Monte Carlo methods*. Vol. 1: Basics. Wiley, New York.
- Nozick, L.K., Turnquist, M.A., Jones, D.A., Davis, J.R., and Lawton, C.R. (2005). Assessing the performance of interdependent infrastructures and optimising investments. *Int. J. Critical Infrastructures*, 1(2/3), 144–154.
- Sallak, M., Schn, W., Aguirre F. (2013). Reliability assessment for multi-state systems under uncertainties based on the Demster-Shafer theory. *IIE Transactions*, 45(9), 995-1007.
- Zio, E. (2009). *Computational methods for reliability and risk analysis*. Series on Quality, Reliability and Engineering Statistics, Vol 14. Singapore: World Scientific Publishing Co. Pte. Ltd.
- Zio, E. (2013). *The Monte Carlo Simulation Method for System Reliability and Risk Analysis*. Series in Reliability Engineering, Springer, London.